

MTA Számítástechnikai és Automatizálási Kutató Intézet Budapest



MAGYAR TUDOMÁNYOS AKADÉMIA
SZÁMITÁSTECHNIKAI ÉS AUTOMATIZÁLÁSI KUTATÓ INTÉZETE

NEMLINEÁRIS EGYENLETRENDSZEREK
MEGOLDÁSI MÓDSZEREI

IRTA:

Hajnal Andrásné

Tanulmányok 38/1975

A kiadásért felelős:

Dr. Vámos Tibor

az

MTA Számítástechnikai és Automatizálási

Kutató Intézet

igazgatója

Jelen dolgozat a "Lineáris és nemlineáris
egyenletrendszerek, sajátértékproblémák" c.
intézeti alapkutatási téma keretében készült.

Beérkezett: 1975. II. 26.

TARTALOMJEGYZÉK

	Oldal
1. BEVEZETÉS.....	5
2. NEMLINEÁRIS EGYENLETRENDSZEREK MEGOLDÁSÁRA SZOLGÁLÓ ITERÁCIÓS MÓDSZEREK ÁTTEKINTÉSE	7
2.1 Newton-módszer és annak bizonyos variációi	8
2.2 Szelő módszerek	13
2.3 Általánosított lineáris módszerek	21
2.4 A gradiens-módszer és módosításai, és a Powell-féle hibrid módszer	24
2.5 Folytatásos módszerek	31
3. NEMLINEÁRIS EGYENLETRENDSZEREK MEGOLDÁSÁRA KÉSZÜLT PROGRAMCSOMAG	36
4. PRÓBAFELADATOK	41

1. BEVEZETÉS

Az utóbbi időben az egész világon előtérbe került a nemlineáris egyenletrendszerek közelítő megoldásának a problematikája. Ez annak tudható be, hogy ezek a módszerek általában csak nagyteljesítményű elektronikus számológépen alkalmazhatók a gyakorlatban, tehát elméleti vizsgálatuk csak ezeknek a gépeknek a megjelenésével vált időszerűvé. Mivel a téma rendkívül szétágazó és nagyon nehezen kezelhető, még nagyon sok kérdés maradt nyitva és egyáltalán nem sikerült az irodalomban semmi utalást találni ezeknek a kérdéseknek a tisztázására. Ilyen tisztázatlan kérdés például a következő: Az egydimenziós nemlineáris egyenletek esetében a polinomokat teljesen külön lehet kezelni a transzcendens egyenletektől. A polinom fogalmát igen könnyen lehet n dimenzióra általánosítani és ezeknek az általánosított polinomoknak a vizsgálata nagyon sok érdekes kérdést vethet fel és izgalmas eredményeket ígér. Ebben a témában már van néhány kezdeti eredmény, de ezek a gyakorlatban nem nagyon használhatóak. Az ebben a tanulmányban ismertetésre kerülő programcsomagban az általánosított polinomokat speciálisan kezeltük és a programozásnál erősen kihasználtuk polinom alakjukat.

Egy másik teljesen tisztázatlan terület a komplex gyökök kérdése. A programcsomagban csak valós gyök keresésével foglalkozunk, de készült egy program komplex gyök keresésére is Newton-módszerrel. A program komplex értékű kezdeti értékeket is keres a Newton-módszer számára.

Annak érdekében, hogy az iterációs eljárások numerikus viselkedéséről értelmes képet alkothassunk igen nagy mennyiségű számítást kellene végrehajtani, amelyben a megoldandó egyenletrendszerek dimenzióját szisztematikusan változtatjuk és minden esetben nagyon sok fajta kezdeti értékrendszerből indulunk ki. Bizonyos mennyiségű számítást végrehajtottunk 2,3,4 és 5 dimenzióra, egyes esetekben több kezdetiértékrendszerből kiindulva, ezekből már le lehet vonni bizonyos következtetéseket például arra vonatkozóan, hogy a számítási hibák hogyan befolyásolják a különböző módszereket, de ennél sokkal több számításra lenne még szükség ahhoz, hogy a

nyitott kérdéseket megválaszolhassuk. Tudomásom szerint ilyen jellegű kísérleteket, melyben elég nagyméretű számítászt hajtottak volna végre ahhoz, hogy már értelmes következtetéseket lehessen levonni, még sehol a világon nem valósítottak meg.

Ebben a tanulmányban megpróbáltam összefoglalni azokat az eljárásokat, amiket nemlineáris egyenletrendszerekre eddig kidolgoztak, konvergenciára vonatkozó tételek és bizonyítások nélkül. A tételeket és bizonyításokat a tanulmány terjedelmi határai miatt nem lehet itt közölni, az érdeklődő olvasó az idézett irodalomban megtalálhatja a meglévő konvergencia eredményeket és bizonyításokat.

A tanulmány tartalmazza még a CDC 3300-as gépre USASI FORTRAN nyelven írt nemlineáris egyenletrendszerek megoldására szolgáló programcsomag logikai diagrammját és leírását. A programcsomag részletes kezelési utasítása a CDC 3300 Felhasználói ismertető 6. számában található meg.

Az utolsó fejezet a programcsomag kipróbálásakor kapott számítási eredményeket tartalmazza.

A nemlineáris egyenletrendszerek megoldási módszereinek egy teljes új osztályával, pontosabban azokkal a módszerekkel, amelyek általánosított inverzeket használnak, itt egyáltalán nem foglalkozunk, mert a programcsomagban ilyen eljárást nem építettünk be.

2. NEMLINEÁRIS EGYENLETRENDSZEREK MEGOLDÁSÁRA SZOLGÁLÓ ITERÁCIÓS MÓDSZEREK ÁTTEKINTÉSE

$Fx = 0$ alakú egyenletrendszerek megoldásával fogunk foglalkozni, ahol $F: D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ alakú leképezés, tehát olyan leképezés, ami az \mathbb{R}^n tér egy D résztartományát az \mathbb{R}^n térre képezi le, $x \in \mathbb{R}^n$.

Egyenletrendszerünket koordináták segítségével a következőképpen lehet leírni:

$$f_1(x_1, x_2, \dots, x_n) = 0$$

$$f_2(x_1, x_2, \dots, x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2, \dots, x_n) = 0$$

ahol f_i valamilyen tetszőleges függvénye a változónak. Az ilyen egyenletrendszerek megoldására általában iterációs módszereket használunk. Iterációs módszereknek a következő típusú eljárásokat nevezzük:

Felvesszük az x^* megoldásnak valamilyen $x^0 \in \mathbb{R}^n$ ismert közelítését. x^0 segítségével valamilyen algoritmussal kiszámítottunk egy x^1 pontot, stb. Ha az x^k pontot már kiszámítottuk, akkor ennek segítségével számítjuk ki az x^{k+1} -eket. Azt mondjuk, hogy az iterációs módszer konvergál, ha

$$\lim_{k \rightarrow \infty} \{x^k\} = x^*$$

Ebben a részben át fogjuk tekinteni a nemlineáris egyenletrendszerek megoldására szolgáló iterációs módszereket, tehát azokat az algoritmusokat, amelyek használatosak arra, hogy az x^k közelítésből megkaphassuk az x^{k+1} közelítést.

2.1 Newton-módszer és annak bizonyos variációi

Az egy dimenziós Newton-módszer általánosan ismert, de a több dimenziós általánosítás kedvéért röviden összefoglaljuk a lényegét.

Legyen f egy egy ismeretlenes valós függvény, és legyen x^* egy gyökhelye. Akkor az ún. párhuzamos hurok módszere abból áll, hogy az x^* valamilyen x^0 közelítésénél az f függvényt az

$$l(x) = \alpha(x - x^0) + f(x^0)$$

lineáris függvényvel helyettesítjük, valamilyen $\alpha \neq 0$ megfelelően választott hajlásszöggel, és ennek a lineáris függvénynek az x^1 gyökét vesszük a következő közelítésnek. Ha ezt az eljárást folytatjuk, akkor a következő iterációs eljárást kapjuk:

$$x^{k+1} = x^k - \alpha^{-1} f(x^k), \quad k = 0, 1, \dots$$

Most ha α -nak az $f'(x^k)$ -t választjuk, akkor kapjuk meg az egy dimenziós Newton-módszert.

A párhuzamos hurok módszerét ki lehet terjeszteni az $F: D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ függvényre, ha α -t egy konstans, nem-szinguláris A mátrixszal helyettesítjük. Ennek megfelelően az n -dimenziós párhuzamos hurok módszerét az

$$x^{k+1} = x^k - A^{-1} F x^k, \quad k = 0, 1, \dots$$

iterációs képlet definiálja, ahol x^i az \mathbb{R}^n n -dimenziós vektortérnek egy eleme.

Ez az iterációs képlet ekvivalens azzal, hogy az F -et az x^k helyen az

$$L_k x = A(x - x^k) + F x^k$$

affin függvényvel helyettesítjük és x^{k+1} -et egyenlővé tesszük az $L_k x = 0$ lineáris egyenletrendszer egyetlen megoldásával.

Geometrialilag ez azt jelenti, hogy x^{k+1} a

$$\sum_{j=1}^n a_{ij}(x_j - x_j^k) + f_i(x^k) = 0, \quad i = 1, \dots, n$$

n -dimenziós hipersíkok metszete az R^{n+1} térben az $x=0$ hipersíkkal.

A fenti iteráció alkalmazásának a kritikus pontja természetesen az A mátrix megválasztása. Ha az egy dimenziós Newton-módszer n -dimenzióra való általánosítását akarjuk megkapni, akkor annak analógiájára A -t $F'(x^k)$ -nek kell megválasztani, ahol $F'(x^k)$ az F leképezés Gateaux (vagy röviden G) deriváltja az x^k pontban. Az F leképezés Gateaux deriváltját a következőképpen definiáljuk:

Definíció:

Egy $F: D \subset R^n \rightarrow R^n$ leképezés Gateaux deriválható a D tartomány egy x belső pontjában, ha létezik egy

$$A \in L(R^n, R^n)$$

lineáris operátor úgy, hogy bármely $h \in R^n$ -re

$$\lim_{t \rightarrow 0} (1/t) \| F(x+th) - Fx - tAh \| = 0$$

Az $F'(x)$ G -deriváltját a következőképpen lehet előállítani az F f_1, f_2, \dots, f_n függvény komponenseinek a segítségével:

Legyen $A = (a_{ij})$ és h -t úgy válasszuk meg, hogy legyen egyenlő e -vel azaz azzal a vektorral, amelynek minden koordinátája 0, kivéve a j -ediket, amely 1-gyel egyenlő, akkor a definícióból következik, hogy

$$\lim_{t \rightarrow 0} (1/t) | f_i(x+te^j) - f_i(x) - ta_{ij} | = 0,$$

tehát mindegyik f_i -nek léteznek az x pontban a parciális derivált-

jai és, hogy

$$\partial_j f_i(x) = \frac{\partial f_i(x)}{\partial x_j} = a_{ij}, \quad i, j = 1, \dots, n$$

Ezért az $F'(x)$ mátrix reprezentációja az ugynevezett Jacobi-féle mátrix,

$$F'(x) = \begin{pmatrix} \partial_1 f_1(x) & \dots & \partial_n f_1(x) \\ \vdots & & \vdots \\ \partial_1 f_n(x) & \dots & \partial_n f_n(x) \end{pmatrix}$$

Az A mátrix megválasztása a következő okokból a legtermészetesebb: az A mátrix megválasztásának egyik feltétele az, hogy a kapott iteráció legalább lokálisan konvergens legyen. Ez azt jelenti, hogy ha x^0 megfelelően közel van az $Fx = 0$ egy x^* megoldásához, akkor

$$\lim_{k \rightarrow \infty} x^k = x^* \quad \text{legyen.}$$

Ennek szükséges és elégséges feltétele, ha $F'(x)$ létezik, hogy

$$\rho(I - A^{-1} F'(x^*)) < 1 \quad \text{legyen,}$$

ahol ρ az A mátrix spektrál sugarát jelöli. Még azt is meg lehet mutatni, hogy minél kisebb ez a ρ , annál gyorsabb a konvergencia. Ebből világosan látszik, hogy A optimális megválasztása az $A = F'(x^*)$. De mivel $F'(x^*)$ -ot nem ismerjük, A -t úgy válasszuk meg, hogy lépésenként változzon és $\lim_{k \rightarrow \infty} A_k = F'(x^*)$ legyen.

A Newton-féle iterációs módszer fontossága abban a tényben rejlik, hogy F -re vonatkozó bizonyos természetes feltételek mellett a következő alakú becslés érvényes rá:

$$\|x^{k+1} - x^*\| \leq C \|x^k - x^*\|^2,$$

feltéve, ha x^k megfelelően közel van az x^* megoldáshoz. Ez azt jelenti, hogy a $(k+1)$ -edik hiba a k -adik hiba négyzetével arányos, tehát a konvergencia igen gyors, ha már x^k közel van az x^* megoldáshoz. Ez az úgynevezett "négyzetes konvergencia" tulajdonsága a többi iterációs módszernek már általában nincs meg, tehát ezért, és az iteráció egyszerűsége és eleganciája miatt a Newton-módszer a nemlineáris egyenletrendszerek megoldásának leginkább tanulmányozott és legjobban kifejlesztett módszere.

Annak ellenére, hogy a Newton-módszer elméletileg a legvonzóbb tulajdonságokkal rendelkezik, a gyakorlatban néha nagyon bonyolult végrehajtani. Minden lépésben meg kell oldani egy $n \times n$ -es lineáris egyenletrendszert, ami ha nagyméretű feladatról van szó, igen nehéz dolog lehet. A második és nagyobb nehézség pedig abban áll, hogy minden iterációs lépésnél ki kell számítani az egyenletrendszer Jacobi mátrixának n^2 komponensét, és ezért hacsak a parciális deriváltaknak nincs valami nagyon egyszerű függvény alakja, ezt jobb elkerülni. Ezért a Newton-módszert leginkább olyan irányokba próbálják módosítani, ahol nem kell explicit formában előállítani a Jacobi mátrixot. A legegyszerűbb út $F'(x)$ kiszámításának elkerülése felé, hogy a parciális deriváltakat differencia hányadosokkal közelítjük. Kétféle differencia közelítést használnak a leggyakrabban:

$$a./ \quad \partial_j f_i(x) \approx (1/h_{ij}) \left[f_i\left(x + \sum_{k=1}^n e^k\right) - f_i\left(x + \sum_{k=1}^{j-1} h_{ik} e^k\right) \right],$$

és

$$b./ \quad \partial_j f_i(x) \approx (1/h_{ij}) \left[f_i(x + h_{ij} e^j) - f_i(x) \right]$$

ahol h_{ij} adott diszkretizálási paraméter, e^j pedig a j -edik egységvektor.

Tehát a Newton-módszer a legkézenfekvőbb általánosítása a következő iterációs eljárás:

Ha $h \in \mathbb{R}^p$ egy paraméter vektor és $\Delta_{ij}(x, h)$ jelöli a $\partial_j f_i(x)$ parciális derivált valamilyen közelítését, ami kielégíti a

$$\lim_{h \rightarrow 0} \Delta_{ij}(x, h) = \partial_j f_i(x), \quad i, j = 1, \dots, n$$

összefüggést, akkor ha $J(x, h)$ -val jelöljük a $\Delta_{ij}(x, h)$ differencia mátrixot, akkor az

$$x^{k+1} = x^k - J(x^k, h^k)^{-1} F x^k, \quad k = 0, 1, \dots,$$

iterációs eljárást diszkretizált Newton-módszernek nevezik.

A $h_{ij} = h$ számára a legegyszerűbb választás $h_{ij} \equiv h$. Ekkor meg lehet mutatni, hogy a konvergencia sebessége lineáris lesz. Ha meg akarjuk közelíteni a Newton-módszer gyors konvergenciáját, akkor ennek szükséges feltétele az, hogy $\lim_{k \rightarrow \infty} h^k = 0$ legyen. Ezt sokféleképpen el lehet érni, például úgy, hogy h^k -t $\gamma_k h$ -nak választjuk valamilyen fix h -val és a $\{\gamma_k\}$ sorozatot pedig úgy választjuk meg, hogy $\lim_{k \rightarrow \infty} \gamma_k = 0$ legyen.

A legérdekesebb módszereket akkor kapjuk, ha a h^k -kat az iterált értékek valamilyen függvényeként állítjuk elő. Ennek a megválasztási módnak az egyik legérdekesebb esete az ún. Steffensen-módszer, amelyről a szelő módszereknel fogunk részletesen beszélni.

A Newton-módszerre általában nem teljesül az ún. "norma-redukáló" tulajdonság, azaz hogy

$$\|F x^{k+1}\| \leq \|F x^k\|, \quad k=0, 1, \dots,$$

valamilyen R^n -beli normában.

Nagyon érdekes variációit kaphatjuk meg a Newton-módszernek akkor is, ha az iterációs képletet átalakítjuk olymódon, hogy ez a tulajdonság teljesüljön. Például a legegyszerűbb módosítás a következő lehet:

$$x^{k+1} = x^k - \omega_k F'(x^k)^{-1} F x^k, \quad k = 0, 1, \dots,$$

ahol az ω_k szorzókat úgy választjuk meg, hogy a normaredukáló tulajdonság teljesüljön. Meg lehet adni elégséges feltételeket

az egyenletrendszerhez, hogy ilyen ω_k -k létezzenek, de ezekkel itt nem foglalkozunk.

Egy másik ilyen módosítás a következő lehet:

$$x^{k+1} = x^k - [F'(x^k) + \lambda_k I]^{-1} Fx^k, \quad k = 0, 1, \dots,$$

Itt a λ_k paramétereket kell úgy megválasztani, hogy a $F'(x^k) + \lambda_k I$ mátrix ne legyen szinguláris még akkor sem, ha maga az $F'(x^k)$ szinguláris és még a norma redukáló tulajdonság is teljesüljön. Ilyen jellegű módszer a Powell-módszer, ami tárgyalásunkban fontos szerepet játszik és később részletesen fogunk vele foglalkozni.

2.2 Szelő módszerek

Az n dimenziós szelő módszereket a legegyszerűbben úgy lehet megkapni, hogy az egy-dimenziós szelő módszer n -dimenzióra való általánosításaiként tekintjük őket.

Az egy dimenziós szelő módszer lényege a következő. Ismerjük az $f(x) = 0$ nemlineáris egyenlet valamilyen x^* valós gyökének egy x^k közelítést. Akkor az x^{k+1} -edik közelítést úgy kapjuk meg, hogy az $f(x)$ -et az

$$l(x) = \left[\frac{f(x^k + h^k) - f(x^k)}{h^k} \right] (x - x^k) + f(x^k)$$

lineáris függvénnyel helyettesítjük, és x^{k+1} az $l(x) = 0$ lineáris egyenletnek a megoldása. Itt tulajdonképpen két fontos esetet szokás megkülönböztetni a h^k megválasztásától függően:

- a./ $h^k = \bar{x} - x^k$, ahol \bar{x} valamilyen fix pont. Ekkor kapjuk meg az ismert regula falsi módszert.
- b./ $h^k = x^{k-1} - x^k$, ezt egy dimenziós szelő módszernek nevezik.

Az $l(x)$ tulajdonképpen az $f(x)$ lineáris interpolációja az x^k és

$x^k + h^k$ pontok között. Ezt az eljárást nagyon könnyű n -dimenzióra általánosítani. Egyszeűen mindegyik f_i , $i=1, \dots, n$ komponens felületet egy olyan hipersikkal helyettesítjük, amely adott $n+1$ darab $x^{k,j}$, $j=0,1, \dots, n$ ponton az f_i -t interpolálja az x^k pont környezetében. Azaz találni kell olyan a_i vektort és olyan α_i skalárt, hogy az $L_i x = \alpha_i + x^T a_i$ affin leképezés kielégítse az

$$L_i x^{k,j} = f_i(x^{k,j}), \quad j = 0,1, \dots, n$$

összefüggést.

A következő iterációs értéket, x^{k+1} -et úgy kapjuk meg, hogy ennek az n darab R^{n+1} -beli hipersiknak a metszésvonalát vesszük az $x=0$ hipersikkal, azaz x^{k+1} az $L_i x = 0$, $i = 1, \dots, n$ lineáris egyenletrendszernek a megoldása lesz.

Ez az általános szelő módszer, most az $x^{k,j}$ interpolációs segéd-pontok megválasztásától függően rengeteg különböző módszert nyerhetünk.

Annak érdekében, hogy az n -dimenziós szelő módszereket kicsit részletesebben tárgyalhassuk, két definícióra van szükségünk.

1. Definíció

Bármely $n+1$ darab x^0, \dots, x^n R^n -beli pont általános helyzetben van, ha az $x^0 - x^j$, $j = 1, \dots, n$ vektorok lineárisan függetlenek.

2. Definíció

Legyen $F: D \subset R^n \rightarrow R^n$, azaz F egy olyan leképezés, amely az R^n tér valamilyen D tartományát a teljes R^n térre képezi le és tegyük fel, hogy az $x^0, \dots, x^n \in D$ pontok és az Fx^0, \dots, Fx^n pontok általános helyzetben vannak.

Akkor az

$$x^s = -A^{-1}a \quad \text{pontot,}$$

ahol a és A kielégítik az

$$a + Ax^j = Fx^j, \quad j = 0, \dots, n$$

összefüggést, alap szelő közelítésnek nevezzük az x^0, \dots, x^n -re vonatkozóan.

A második definícióból következik, hogy ahhoz, hogy kiszámítsunk egy alap szelő közelítést, először meg kell találni a -t és A -t, amely kielégíti az $a + Ax^j = Fx^j$ összefüggést. Ehhez viszont meg kell oldani a lineáris egyenletrendszert a -ra és A -ra és utána meg kell oldani az $a + Ax = 0$ lineáris egyenletrendszert.

A továbbiakban kiderül, hogy az $a + Ax$ interpolációs függvény explicit kiszámítására egyáltalán nincs szükség. Két olyan alternatív megfogalmazást is fogunk mutatni, ahol csak egy lineáris egyenletrendszert kell megoldani.

a./ Wolfe-féle szelő formula

Legyenek az x^0, \dots, x^n pontok és az Fx^0, \dots, Fx^n pontok általános helyzetben. Akkor az alap szelő közelítés kielégíti az

$$x^s = \chi z = \sum_{j=0}^n z_j x^j$$

összefüggést, ahol a $z = (z_0, \dots, z_n)^T$ vektor az

$$\begin{pmatrix} 1 & \dots & 1 \\ Fx^0 & & Fx^n \end{pmatrix} z = (1, 0, \dots, 0)^T$$

$(n+1) \times (n+1)$ -es lineáris egyenletrendszer egyetlen megoldása.

b./ Newton-féle szelő formula

Vezessük be a következő operátort:

$$J : D_k \subset \mathbb{R}^n \times L(\mathbb{R}^n) \rightarrow L(\mathbb{R}^n)$$

ahol $L(\mathbb{R}^n)$ az \mathbb{R}^n téren értelmezett lineáris operátorok tere, $\mathbb{R}^n \times L(\mathbb{R}^n)$ az \mathbb{R}^n és $L(\mathbb{R}^n)$ tér Descartes-szorzata, és

$$J(x, H) = (F(x + He^1) - Fx, \dots, F(x + He^n) - Fx) H^{-1}$$

ahol, ha D az F értelmezési tartománya, akkor

$$D_k = \left\{ (x, H) \mid x + He^i \in D, i = 1, \dots, n; H \text{ nem szinguláris} \right\}$$

Legyenek az x^0, \dots, x^n és Fx^0, \dots, Fx^n pontok általános helyzetben, és legyen

$$H = (x^1 - x^0, \dots, x^n - x^0)$$

Akkor $J(x^0, H)$ nem szinguláris és az x^s alap szelő közelítést a következőképpen kapjuk meg:

$$x^s = x^0 - J(x^0, H)^{-1} Fx^0$$

Meg kell jegyezni, hogy ha bevezetjük a

$\Gamma = (Fx^1 - Fx^0, \dots, Fx^n - Fx^0)$ jelölést, akkor a Newton-féle szelő formulát a következő alakban is írhatjuk:

$$x^s = x^0 - H \Gamma^{-1} Fx^0$$

Mindkét szelő formula megfogalmazásából következik, hogy az alap szelő közelítés kiszámításához csak egyetlen lineáris egyenletrendszert kell megoldani, és utána ki kell számítani

az x^0, \dots, x^n vektorok egy lineáris kombinációját. Mindkét szelő formulát igen egyszerűen lehet bizonyítani, de itt nem közöljük a bizonyításokat, mert ennek a tanulmánynak csak a módszerek összefoglalása a célja. A bizonyítások az [9] irodalomban megtalálhatók.

A Newton-féle megfogalmazás segítségével most már leírhatjuk az általános szelő módszert a következő igen rövid alakban:

$$\begin{cases} x^{k+1} = x^k - J(x^k, H_k)^{-1} F x^k, & k = 0, 1, \dots \\ H_k = (x^{k,1} - x^k, \dots, x^{k,n} - x^k) \end{cases}$$

ahol

$$x^{k,0} = x^k$$

A szelő módszereknek igen fontos problémája az interpolációs segédpontok megválasztása.

Egy igen egyszerű megválasztási mód a következő:

$$x^{k,j} = x^k + (x_j^{k-1} - x_j^k) e^j, \quad j = 1, \dots, n \quad (1)$$

Ekkor az általános formulában szereplő H_k mátrix a következő alakú lesz:

$$H_k = \text{diag} (x_1^{k-1} - x_1^k, \dots, x_n^{k-1} - x_n^k),$$

tehát egy olyan mátrix, ahol a fődiagonálisban a fenti elemek állnak, az összes többi elem pedig 0.

Ha a segédpontokat a következőképpen válasszuk:

$$x^{k,j} = x^k + \sum_{i=1}^j (x_i^{k-1} - x_i^k) e^i, \quad j = 1, 2, \dots, n \quad (2)$$

akkor az iterációs képletbe behelyettesítve, az így kapott

$$J(x, h) = h_1^{-1} [F(x + h_1 e^1) - Fx], \dots, h_n^{-1} [F(x + \sum_{j=1}^n h_j e^j) - F(x + \sum_{j=1}^{n-1} h_j e^j)]$$

kifejezést, éppen a diszkretizált Newton-módszer iterációs képletét kapjuk meg, a 2.1. paragrafusban a./ pont alatt szereplő differencia közelítéssel.

Az előbb említett esetekben a segédpontok csak az x^k és x^{k-1} közelítéstől függtek. Az így kapott módszereket általában szekvenciális két-pontos módszereknek nevezzük.

A segédpontok megválasztásának még rengeteg érdekes esete lehetséges, például meg lehet őket úgy is választani, hogy ne kettő, hanem több előző közelítéstől függjenek, vagy hogy nem szekvenciálisan választjuk meg a pontokat, hanem valamilyen más kritérium szerint, például az előző közelítésekből azt az $n+1$ pontot választjuk, amelyekre $\|Fx^j\|$ a legkisebb.

Itt részletesebben még az $(n+1)$ pontos szekvenciális módszerrel foglalkozunk, mert ennek a megválasztási módnak nagyon sok előnyös tulajdonsága van.

Mint az elnevezésből is következik, az előző $n+1$ közelítést választjuk interpolációs segédpontnak. Ekkor minden új iterációs lépésben csak egyetlen új helyen, a legutolsó közelítés helyén kell kiszámítani a függvényt, lévén hogy a többi pontokban már az előző lépésekben kiszámítottuk. Az első lépést kivéve persze, mert akkor $n+1$ pontban ki kell számítani a függvényértékeket.

A másik nagy számítási megtakarítás abból adódik, hogy a módszer alkalmazása közben fellépő lineáris egyenletrendszert csak egyszer, az első lépésben kell megoldani. Ez a következők

miatt lehetséges: A Newton-féle szelő formulánál láttuk, hogy az iterációs képlet a következő alakban is írható:

$$x^{k+1} = x^k - H_k \Gamma_k^{-1} Fx^k,$$

ahol

$$H_k = (x^k - x^{k-1}, x^{k-1} - x^{k-2}, \dots, x^{k-n+1} - x^{k-n}) \quad \text{és}$$

$$\Gamma_k = (Fx^k - Fx^{k-1}, \dots, Fx^{k-n+1} - Fx^{k-n})$$

Akkor ha feltesszük, hogy a Γ_p és Γ_{p+1} mátrixok $k=p, p+1$ -re nem szingulárisak és jelöljük a Γ_p^{-1} mátrix sorait v^1, \dots, v^n -el, akkor

$$\Gamma_{p+1}^{-1} = B - \frac{B(q^p - q^{p-n})v^n}{1 + v^n(q^p - q^{p-n})},$$

ahol $q^i = Fx^{i+1} - Fx^i$ és B az a mátrix, amelynek a sorai v^n, v^1, \dots, v^{n-1} .

A fenti képlet bizonyítása nagyon egyszerű és egyszerűen abból következik, hogy az invertálandó mátrix minden lépésben csak egy diáddal módosul, tehát alkalmazhatjuk rá az ismert Sherman-Morrison féle formulát.

Látjuk tehát, hogy az $(n+1)$ -lépéses szelő módszer követeli meg a legkevesebb számolást, tehát számítástechnikai szempontból igen előnyösnek látszik. Sajnos azonban igen instabil, és semmilyen kielégítő konvergencia eredményt nem lehet rá bizonyítani. Ezzel szemben az előzőekben bemutatott kétpontos módszerek megőrzik a Newton-módszer alapvető tulajdonságait és nagyon jó lokális konvergencia tulajdonságokat lehet rájuk bizonyítani.

Végül bemutatjuk az iterációs módszereknek egy kevésbé ismert osztályát, amelyek szoros kapcsolatban állnak a szelő módszerekkel. Ezek az ún. Steffensen-módszerek.

Az egy dimenziós Steffensen-módszer iterációs képlete a következő:

$$x^{k+1} = x^k - \frac{f(x^k)}{f(x^k + f(x^k)) - f(x^k)} \cdot f(x^k), \quad k = 0, 1, \dots$$

Ez az iteráció nagyon érdekes, mivel megfelelő feltételek mellett ugyanugy négyzetesen konvergál, mint a Newton-módszer, ugyanakkor nem szükséges hozzá a függvény deriváltja.

Az iterációs képletből észre lehet venni, hogy ez lényegében azonos az általános szelő módszer iterációs képletével $h^k = f(x^k)$ helyettesítés mellett. Ebből viszont azonnal adódik az n dimenzióra szóló általánosítás.

Például a két pontos szelő módszer analógiájára, az első fajta alappont megválasztással a következő Steffensen módszert kapjuk:

$$x^{k+1} = x^k - J(x^k, Fx^k)^{-1} Fx^k \quad (3)$$

ahol J azonos a szelő módszernél definiált J leképezéssel.

Természetesen ugyanugy meg lehet kapni a megfelelő Steffensen módszert a második fajta két pontos alappont megválasztásához is, csak a h_i -k helyébe Fx_i^k -t kell helyettesíteni $i = 1, \dots, n$ -re.

2.3. Általánosított lineáris módszerek

Az előzőekben leírt Newton és szelő módszerek egy ismeretlenes nemlineáris egyenletre vonatkozó megoldási módszerek több dimenzióra való általánosításai voltak. Ebben a részben olyan módszerekkel fogunk foglalkozni, amelyek lineáris egyenletrendszerek megoldására való iterációs módszerek nemlineáris esetre való általánosításai. Ezeknek a módszereknek az az előnye az előzőekben szereplő módszerekkel szemben, hogy lényegesen kevesebb a memória igényük, ugyanis nem kell minden lépésben megoldani egy lineáris egyenletrendszert, hátrányuk viszont az, hogy nagyon ritkán, csak egészen speciális alakú egyenletrendszerekre konvergálnak, pontosabban azokra az egyenletrendszerekre, amelyeknek a Jacobi-féle mátrixa kielégíti azokat a feltételeket, amelyeket a módszer a megfelelő lineáris egyenletrendszer mátrixára elégséges feltételként megkövetel. Például a Gauss-Seidel féle iterációnál a mátrix fődiagonálisában álló elemek abszolút értékének nagyobb vagy egyenlőnek kell lenni, mint a megfelelő sorban álló összes többi elem abszolút értékének összege, amíg nagyon szigorú feltétel és nagyon kevés egyenletrendszernél teljesül.

A legismertebb iterációs módszer az $Ax = b$ alakú lineáris egyenletrendszerek megoldására a Gauss-Seidel féle módszer. Ennél az eljárás a következő:

Tegyük fel, hogy az A együttható mátrix fődiagonálisában álló a_{ii} elemek közül egyik sem 0. Akkor ha feltesszük, hogy már ismerjük az $x^k = (x_1^k, \dots, x_n^k)$ -edik iteráltat és az x^{k+1} , vagyis a $(k+1)$ -edik iterált első $i-1$ -edik elemét, akkor a $k+1$ -edik iterált i -edik elemét úgy határozzuk meg, hogy megoldjuk az i -edik egyenletet x_i -re mint egyetlen ismeretlenre úgy, hogy az első $i-1$ -edik ismeretlen helyére a $(k+1)$ -edik iterált már kiszámított értékeit az $(i+1)$ -től n -ig lévő ismeretlenek helyére pedig a k -edik iterált megfelelő értékeit helyettesítjük. Azaz megoldjuk a következő egyenletet

$$\sum_{j=1}^{i-1} a_{ij} x_j^{k+1} + a_{ii} x_i + \sum_{j=i+1}^n a_{ij} x_j^k = b$$

x_i -re és ez lesz x_i^{k+1} .

A módszer nemlineáris egyenletrendszerekre teljesen azonos ezzel, azzal a különbséggel, hogy az egyismeretlenes egyenlet, amit minden lépésben meg kell oldani egy

$$f_i(x_1^{k+1}, \dots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \dots, x_n^k) = 0, \quad i = 1, \dots, n$$

alakú nemlineáris egyenlet lesz.

A gyakorlatban a Gauss-Seidel iteráció helyett egy fontos módosítást szokás használni, az ún. successiv overrelaxációs módszert, vagy röviden SOR módszert, ami lényegében azonos vele, azzal a módosítással, hogy nem magát az x_i megoldást veszi x_i^{k+1} -nek, hanem a következőt:

$$x_i^{k+1} = x_i^k + \omega(x_i - x_i^k)$$

ahol az $1 \leq \omega \leq 2$ valós számot relaxációs paraméternek nevezzük. $\omega = 1$ -re az eredeti Gauss-Seidel iterációt kapjuk vissza.

A módszer végrehajtása közben adódó nemlineáris egyenlet megoldására természetesen bármilyen nemlineáris egyenletekre alkalmazható módszert választhatunk. Ennek a másodlagos iterációs módszernek a megválasztásától függően igen sokféle SOR iterációról beszélhetünk. Például ha a Newton-módszert választjuk, megkapjuk az SOR-Newton iterációt, stb.

Természetesen az elsődleges iteráció számára is többféle általánosított lineáris iteráció áll a rendelkezésünkre. Ezek közül még a Jacobi-féle iterációról fogunk megemlékezni itt és a Gauss-Seidel iteráció egy másik módosításáról a Seidel-módszerről.

A nemlineáris Jacobi iteráció k -adik lépését úgy definiáljuk, hogy meg kell oldani a következő alakú egyenleteket:

$$f_i(x_1^k, \dots, x_{i-1}^k, x_i, x_{i+1}^k, \dots, x_n^k) = 0 \quad i = 1, 2, \dots, n$$

$$x_i \text{ -re} \quad \text{és} \quad x_i^{k+1} = x_i, \quad i = 1, \dots, n$$

Azaz megoldjuk az i -edik egyenletet x_i -re, míg az összes többi x_j értékét x_j^k -nek tartjuk meg.

A fent leírt Gauss-Seidel iterációt ciklikus Gauss-Seidel eljárásnak nevezzük, mert az $f_i = 0$ egyenleteket a természetes sorrendjükben oldjuk meg. Szokás még ugyanevezett "free-steering" módszereket is használni, amelyekben az egyenleteket lényegében tetszőleges sorrendben oldjuk meg, például véletlen kiválasztással. Egy másik eljárás, aminek lineáris egyenletrendszerekre való alkalmazása már klasszikus, és Seidel-módszernek nevezik az, hogy azt az egyenletet választjuk következőnek, amelyiknél a függvényérték a legnagyobb. Ekkor az iteráció k -adik lépése a következőképpen néz ki:

$$a/ \text{ Válasszuk meg } j\text{-t úgy, hogy } |f_j(x^k)| \leq |f_i(x^k)|, \quad i = 1, \dots, n$$

b/ Oldjuk meg az

$$f_j(x_1^k, \dots, x_{j-1}^k, x_j, x_{j+1}^k, \dots, x_n^k) = 0$$

egyenletet x_j -re és most legyen

$$x^{k+1} = x^k + \omega (x_j - x_j^k) e^j$$

Meg kell jegyeznünk, hogy a fenti előírásoknak csak akkor van értelme, ha a megoldandó egyenleteknek csak egyetlen megoldásuk van a keresett gyök közelében. Azonban lehet F -re olyan feltételeket adni, hogy ez a szükséges feltétel teljesüljön. Természetesen ezek a feltételek nem elégségesek ahhoz, hogy az iterációs módszer konvergáljon is. Azt is meg kell itt még jegyezni, hogy ha

másodlagos iterációnak a Newton-módszert választjuk, akkor elveszítjük azt az előnyt, hogy nem kell kiszámítani a parciális deriváltakat. Ezért a gyakorlatban előnyösebb valamilyen olyan egydimenziós módszert választatni az egyenletek megoldására, amelyik nem használja a függvény deriváltját, tehát pl. szelő módszert, vagy a Steffensen-módszert.

2.4. A gradiens-módszer és módosításai, és a Powell-féle hibrid módszer

Az itt következő módszerek könnyebb tárgyalása céljából tegyük fel, hogy a továbbiakban olyan iterációs módszerekről fogunk beszélni, amelyek a következő alakba írhatók:

$$x^{k+1} = x^k - H^k F x^k t_k,$$

ahol H^k olyan $n \times n$ -es mátrix, amit az alkalmazott módszer határoz meg, a Newton-módszernél például $H^k = F'(x^k)$, t_k pedig skalárfaktor, ami függhet $F x^{k+1}$ -től, de nem feltétlenül. Ha t_k független $F x^{k+1}$ -től, akkor az értéke rendszerint 1, mint például a Newton-módszer esetében, ha függ, akkor általában úgy választják meg, hogy kielégítse a következő egyenlőtlenséget:

$$\|F x^{k+1}\| < \|F x^k\| \quad (1)$$

Ebben a megfogalmazásban a gradiens-módszert, vagy más néven a legmeredekebb csökkenés módszerét a következőképpen írhatjuk le:

Legyen: $H^k = F'(x^k)^T$, $t_k > 0$ és legyen

$$G(x) = F x^T F x$$

Akkor

$$\text{grad } G(x) = 2 F'(x)^T F x$$

Könnyű látni, hogy az $F x = 0$ egyenlet megoldása a $G(x)$ nulla értékű minimumával azonos, tehát ennek megfelelően egyenletünket úgy is meg lehet oldani, hogy $G(x)$ -et minimalizáljuk. Miután $G(x)$ legmeredekebb csökkenésének az iránya a $-\text{grad } G(x)$ irányában van, ebből következik, hogy $G(x)$ -et redukálni lehet ha H^k -nak $F'(x^k)^T$ -t választjuk és t_k -t úgy választjuk meg, hogy kielégítse az (1)

egyenlőtlenséget.

A gradiens módszer abban az esetben mondja fel a szolgálatot, ha a kezdeti érték az Fx egy nem nulla minimumának a környezetében van, miután az algoritmus akkor ehhez a minimumhoz konvergál. Ebben az esetben meg lehet mutatni, hogy

$$F'(x)^T Fx = 0, \quad \text{tehát mivel } Fx \neq 0$$

ebből következik, hogy $F'(x)$ szinguláris. De mivel általában a többi iterációs módszer is felmondja a szolgálatot ha $F'(x)$ szinguláris, ez egyáltalán nem írható a módszer hátrányára.

A gradiens módszerről be lehet látni, hogy csak lineárisan konvergál és a gyakorlatban be is bizonyosodott, hogy nagyon lassu. Tehát mivel ennél a módszernél is szükség van az explicit Jacobi mátrix használatára, akkor inkább a Newton-módszerrel érdemes dolgozni. Ennek a módszernek az egyetlen előnye a Newton-módszerrel szemben az, hogy sok esetben jól használható amikor a Newton-módszer felmondja a szolgálatot, mert valahol a közelítés folyamán szinguláris Jacobi-mátrix fordul elő. Ezért tehát a legjobb a kevert módszert használni, ha nincs jó becslés, tehát elindulni ezzel a módszerrel és ha már van egy durva gyök, akkor áttérni a Newton-módszerre. Ennek a célnak az elérésére azonban lényegesen elegánsabb módok is vannak. Az egyik például a következő:

Levenberg-Marquardt módszere

H^k -t válasszuk a következő alakú kifejezésnek:

$$H^k = (F'(x^k)^T F'(x^k) + \lambda_k I)^{-1} F'(x^k)^T, \quad t_k=1, \quad \lambda_k \geq 0$$

Ebből a kifejezésből látható, hogy ha λ_k nő, akkor a lépéshossz csökken és ezért a lépéshossz vektora a tiszta legmeredekebb csökkenés vektorához konvergál. Meg lehet mutatni, hogy elég nagy λ_k -ra az (1) egyenlőtlenség teljesül és ezért ajánlatos λ_k -t így megválasztani. Másrészt $\lambda_k=0$ -nál a módszer a Newton-módszerre redukálódik, ezért ez a módszer a gradiens és a Newton-módszer bizonyos jó tulajdonságait egyesíti, ha a megoldáshoz közeledve λ_k -t állandóan csökkentjük. További jó tulajdonsága az,

hogy ha $\lambda_k > 0$, akkor mindig létezik az inverz mátrix, tehát a korrekciót mindig végre lehet hajtani. Ez a módszer is akkor mondja fel a szolgálatot, ha Fx egy nem nulla lokális minimumához konvergálunk.

Powell-féle hibrid módszer

A Powell-féle hibrid módszer nagyon hasonló a Levenberg-Marquardt módszerhez. A legfontosabb különbség az közöttük, hogy a Powell-módszer nem kívánja meg a Jacobi-mátrix explicit kiszámítását, hanem egy Powell által kifejlesztett technikát használ a Jacobi mátrix elemeinek numerikus közelítésére az $f_i(x^k)$ ($i = 1, 2, \dots, n$) értékek segítségével. Ez azért sokkal előnyösebb mint a hagyományos módszerek, mert csak $O(n^2)$ műveletet használ iterációs lépésenként a Jacobi mátrix inverzének számítására, míg ha a hagyományos módon minden lépésben megoldjuk a fellépő lineáris egyenletrendszert, az $O(n^3)$ műveletet igényel. A módszer megmutatja a Levenberg-Marquardt iterációnak azt a fontos tulajdonságát, hogy ha a teljes Newton-Raphson korrekció túl nagy, akkor az $x^{(k)}$ -től való eltérést a legmeredekebb csökkenés irányába torzítja.

Hogy az iterációs eljárást könnyen leírassuk, bevezetünk még egy további jelölésbeli rövidítést az általános iteráció definíciójába.

Ha meg akarjuk oldani iterációs módszerrel az

$$Fx = \begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) = 0 \end{cases}$$

nemlineáris egyenletrendszert, akkor feltesszük, hogy ismerünk már valamilyen x^k közelítő megoldást és x^{k+1} -et úgy kapjuk meg, hogy

$$x^{(k+1)} = x^{(k)} + \delta^{(k)}$$

ahol a $\delta^{(k)}$ korrekciót minden módszernél más és más módon számít-

jük ki. Például a Newton-módszernél $\delta^{(k)}$ az

$$F(x^k) + F'(x^k) \delta^k = 0$$

lineáris egyenletrendszer megoldása.

A Powell-módszer algoritmusának az ismertetésénél először feltesz-
szük, hogy ismerjük a Jacobi mátrix elemeinek explicit kifejezését,
majd az algoritmus leírása után ismertetjük a Jacobi mátrix köze-
lítésének technikáját.

Az iterációs eljárás a következő:

Hogy a k -adik iterációt elkezdhessük, szükségünk van a megoldás
egy x^k becslésére, egy $\Delta^{(k)}$ lépéshosszra és két számra E -re és
 M -re. A lépéshosszat minden iterációs lépésben megváltoztatjuk
és a célja az, hogy az $(x^{(k+1)} - x^{(k)})$ módosítás hosszát szabályoz-
za abból a célból, hogy $F(x)$ értéke csökkenjen. Feltéve, hogy
 $F(x)$ lényegesen csökken, még arra is törekedni kell, hogy $\Delta^{(k)}$
értékét elég nagynak tartsuk meg, mert ha $\Delta^{(k)}$ értéke túl kicsivé
válna, túl sok iterációs lépésre lenne szükség. Az E és M
számoknak fix pozitív értéket adunk mielőtt az iterációt elkezde-
nénk. Az eljárás befejeződik, ha az $F(x)$ értéke vagyis R^n -beli
normája kisebb mint E , vagy ha $F(x)$ gradiense olyan kicsivé
válik, hogy x távolsága a megoldástól, valószínűleg nagyobb lesz,
mint M .

A k -adik iterációs lépésben először kiszámítjuk a Jacobi mátrix
elemeit $x^{(k)}$ -nál és utána kiszámítjuk a teljes Newton-Raphson
korrekciót a $\delta^{(k)}$ -t és az $F(x)$ $g^{(k)}$ gradiensét az $x^{(k)}$ helyen.

$$g_j^{(k)} = \left[\frac{\partial}{\partial x_j} F(x) \right]_{x=x^{(k)}} = 2 \sum_{i=1}^n \left[f_i(x) \frac{\partial}{\partial x_j} f_i(x) \right]_{x=x^{(k)}}$$

Utána megvizsgáljuk az

$$F(x^{(k)}) \geq M \|g^{(k)}\|$$

egyenlőtlenséget.

Ha ez az egyenlőtlenség teljesül, akkor abba hagyjuk az iterációt, mert nagy a valószínűsége annak, hogy a sorozat nem megoldáshoz, hanem $F(x)$ egy lokális minimumához konvergál. Ez abból következik, hogy mivel $\|g^{(k)}\|$ az $F(x)$ legmeredekebb csökkenésének iránya $x^{(k)}$ -nél, azaz $x^{(k)}$ változásának hossza, ami szükséges ahhoz, hogy $F(x)$ -et nullára redukáljuk, nagyobb lesz mint M . Ez pedig nem jó, ha M -et az előírt módon írjuk elő. Ebben az esetben $\|g^{(k)}\|$ -nak szokatlanul kicsinek kell lenni, tehát valószínűleg a közelben van $F(x)$ -nek egy stacionárius pontja és ez okozza a nehézséget. Ha a feltétel nem teljesül, akkor kiszámítjuk azt a $d^{(k)}$ korrekciót, amit az $x^{(k)}$ vektorhoz hozzá kell adni. Ez a $d^{(k)}$ a klasszikus $d^{(k)}$ korrekció lesz, ha $\Delta^{(k)} \geq \|d^{(k)}\|$. De ha

$$\Delta^{(k)} < \|d^{(k)}\|, \quad \text{akkor} \quad (1)$$

a $\bar{d}^{(k)}$ hosszát $\Delta^{(k)}$ -nak vesszük fel és a $\bar{d}^{(k)}$ a következő alakú lesz:

$$\bar{d}^{(k)} = \alpha_1 d^{(k)} + \beta_1 g^{(k)} \quad (2)$$

ahol α_1 és β_1 skalárok. Pontosabban $\alpha_1 = 0$ ha az $F(x)$ legmeredekebb csökkenése vektorának irányába tett lépés

$$\bar{d}^{(k)} = -\Delta^{(k)} g^{(k)} / \|g^{(k)}\| \quad (3)$$

nem haladja meg a $g^{(k)}$ vektor irányában lévő jóslt minimumát az $F(x)$ -nek. Ez a jóslt minimum a következő pontban van:

$$x^{(k)} - \left\{ \frac{1}{2} \left(\|g^{(k)}\|^2 / \|J^{(k)} g^{(k)}\|^2 \right) \right\} g^{(k)} \quad (4)$$

ha $J^{(k)}$ jelöli a Jacobi-féle mátrixot.

Most megnézzük, hogy teljesül-e a

$$\Delta^{(k)} \leq \frac{1}{2} \|g^{(k)}\|^3 / \|J^{(k)} g^{(k)}\|^2 \quad (5)$$

egyenlőtlenség.

Ha mind az (1), mind az (5) egyenlőtlenség teljesül, akkor $\bar{\sigma}^{(k)}$ -t a (3)-as egyenlőséggel definiáljuk. Abban az esetben ha az (1) feltétel teljesül, de az (5) nem, akkor az $\{x^{(k)} + \bar{\sigma}^{(k)}\}$ pontot úgy határozzuk meg, hogy azon az egyenes vonalon legyen, ami az $\{x^{(k)} + \sigma^{(k)}\}$ pontot a (4)-ben definiált ponttal összeköti. Ehhez felhasználjuk $\|\bar{\sigma}^{(k)}\| = \Delta^{(k)}$ feltételt.

Most már előirtuk $\bar{\sigma}^{(k)}$ -t minden lehetséges esetre. Az iterációs eljárás következő lépése az, hogy kipróbáljuk az $(x^{(k)} + \bar{\sigma}^{(k)})$ becslést. Tehát először is kiszámítjuk az $f_i(x)$ ($i = 1, 2, \dots, n$) függvényértékeket az új pontban.

Ha az

$$F(x^{(k)} + \bar{\sigma}^{(k)}) < F(x^{(k)})$$

egyenlőtlenség teljesül, akkor megnézzük, hogy az $\|F(x^{(k+1)})\| < \varepsilon$ konvergencia kritérium teljesül-e. Ha nem teljesül, akkor $x^{(k+1)} = x^{(k)}$, és redukáljuk a $\Delta^{(k)}$ lépéshosszat. Ez a következő módon történik:

Képezzük a következő mennyiséget:

$$\Phi^{(k)} = \sum_{i=1}^n \left\{ f_i(x^{(k)}) + \sum_{j=1}^n J_{i,j}^{(k)} \bar{\sigma}_j^{(k)} \right\}^2$$

Ez a mennyiség az $x^{(k)} + \bar{\sigma}^{(k)}$ -nál képzett maradékok négyzetösszegének jóslott értéke, és kisebb mint $F(x^{(k)})$.

Ha úgy találjuk, hogy ez annyira rossz, hogy a négyzetösszegek aktuális értéke kielégíti az

$$F(x^{(k)} + \bar{\sigma}^{(k)}) > (1 - \varepsilon) F(x^{(k)}) + \varepsilon \Phi^{(k)}$$

egyenlőtlenséget, ahol $\xi \in (0,1)$, ebből következik, hogy az $f_i(x)$ függvények Jacobi elemekből kapott lineáris approximációja nem megfelelő a $\|\bar{d}^{(k)}\|$ távolságra, tehát redukálni kell a $\Delta^{(k)}$ lépéshosszt, azaz meg kell szorozni egy $\mu, 0 < \mu < 1$ konstans faktorral. A programcsomagban lévő programban $\xi = 0,1$ és $\mu = 0,5$. Ha az egyenlőtlenség nem teljesül, akkor növelni is lehet $\Delta^{(k)}$ -t. Ezzel az iteráció egy lépése teljesen definiálva van. Most a Jacobi mátrix aktuális számítását ismertetjük.

Bevezetjük a következő jelöléseket:

$$\text{Legyen } \gamma_i^{(k)} = f_i(x^{(k)} + \bar{d}^{(k)}) - f_i(x^{(k)})$$

és jelöljük $J^{(k)}$ -val az $F(x)$ Jacobi mátrixát az $x^{(k)}$ pontban, $H^{(k)}$ -val a $J^{(k)}$ mátrix inverzét.

Akkor érvényesek a következő összefüggések $J^{(k+1)}$ -re és $H^{(k+1)}$ -re:

$$\begin{aligned} J^{(k+1)} &= J^{(k)} + \alpha (\gamma^{(k)} - J^{(k)} \bar{d}^{(k)}) \bar{d}^{(k)T} / \|\bar{d}^{(k)}\|^2 \\ H^{(k+1)} &= H^{(k)} + \alpha \frac{(\bar{d}^{(k)} - H^{(k)} \gamma^{(k)}) \bar{d}^{(k)T} H^{(k)}}{\alpha (\bar{d}^{(k)T} H^{(k)} \gamma^{(k)}) + (1-\alpha) \|\bar{d}^{(k)}\|^2} \end{aligned} \quad (6)$$

ahol α értékét a következő egyenlőtlenség teljesülésétől függően választjuk meg.

$$|(\bar{d}^{(k)T} H^{(k)} \gamma^{(k)})| < 0,1 \|\bar{d}^{(k)}\|^2$$

Ha az egyenlőtlenség teljesül $\alpha = 0,8$, ha nem teljesül akkor $\alpha = 1$.

A (6)-os összefüggések bizonyítását a [2] irodalom tartalmazza.

2.5. Folytatásos módszerek

Az eddig tárgyalt módszerek legnagyobb része általában akkor és csak akkor konvergál az $Fx = 0$ egy x^* megoldásához, ha a kezdeti közelítések megfelelően közel vannak x^* -hoz. Az ebben a részben tárgyalt folytatásos módszereket úgy is lehet tekinteni, mint egy kísérletet arra, hogy a módszerek konvergencia tartományát kiterjesszük, vagy megfelelően közeli kezdőpontot találjunk egy bizonyos eljáráshoz.

Sok, a gyakorlatban előforduló esetben a probléma természetes módon függhet egy olyan t paramétertől, hogy amikor a paraméter valamilyen meghatározott értéket vesz fel, mondjuk 1-et, akkor megkapjuk az F leképezést, míg $t = 0$ -ra az eredményül kapott $F_0 x = 0$ egyenletrendszernek van egy ismert x^0 megoldása. Pontosabban, egy leképezés helyett egy egész leképezés családnak van, $H: D_x [0, 1] \subset \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ úgy, hogy

$$H(x, 0) = F_0 x, \quad H(x, 1) = Fx, \quad \forall x \in D,$$

ahol a $H(x, 0) = 0$ egyenletrendszer x_0 megoldását ismerjük és a $H(x, 1) = 0$ egyenletet kell megoldani.

Ha F nem is függ természetes módon egy megfelelő t paramétertől, akkor is definiálhatunk egy H leképezés családot, amely kielégíti a fenti összefüggéseket.

Ezt például a következőképpen lehet megtenni.

Legyen

$$H(x, t) = tFx + (1-t)F_0 x, \quad x \in D, \quad t \in [0, 1]$$

ahol $F_0 x$ egy olyan leképezés, hogy az $F_0 x = 0$ egyenletrendszer egy x_0 megoldását ismerjük.

Vagy legyen

$$H(x, t) = Fx + (t-1) Fx^0, \quad x \in D, \quad t \in [0, 1],$$

ahol x^0 adott pont R^n -ben.

Megjegyezzük, hogy a második fajta megfogalmazást úgy kaphatjuk meg az elsőből, hogy

$$F_0 x = Fx - Fx^0 \quad -t$$

helyettesítünk.

Most függetlenül attól, hogy hogyan kaptuk meg H -t, tekintsük a következő egyenletet:

$$H(x, t) = 0, \quad t \in [0, 1] \quad (1)$$

és tegyük fel, hogy (1)-nek minden $t \in [0, 1]$ -re van egy $x = x(t)$ megoldása, amely folytonosan függ t -től. Más szavakkal, tegyük fel, hogy létezik egy folytonos $x: [0, 1] \rightarrow D$ leképezés úgy, hogy

$$H(x(t), t) = 0, \quad \forall t \in [0, 1] \quad (2)$$

Ekkor $x(t)$ olyan térgörbét határoz meg R^n -ben, amelynek az egyik végpontja az adott x^0 pont, a másik végpontja pedig az $x^* = x(1)$ megoldása az $Fx = H(x, 1) = 0$ -nak.

Hogy egy ilyen folytonos térgörbe létezését hogyan lehet biztosítani arra vonatkozóan ki lehet mondani a következő tételt:

Tétel:

Legyen $F: R^n \rightarrow R^n$ folytonosan differenciálható R^n -ben és tegyük fel, hogy $F'(x)$ nem szinguláris minden $x \in R^n$ -re és $\|F'(x)^{-1}\|$ korlátos minden $x \in R^n$ -re.

Akkor bármely fix $x^0 \in R^n$ -hez létezik egyetlen $x: [0, 1] \rightarrow R^n$ leképezés úgy, hogy $H(x(t), t) = 0$ legyen minden $t \in [0, 1]$ -re és $H(x, t) = Fx + (t-1)Fx^0$ alakú H leképezésre.

Továbbá x folytonosan differenciálható és

$$x'(t) = -F'(x(t))^{-1} Fx^0, \quad \forall t \in [0,1], \quad x(0) = x^0$$

A tétel bizonyítása [9] -ben megtalálható. A továbbiakban fel fogjuk tenni, hogy F kielégíti a tétel feltételeit, tehát létezik a követelményeknek eleget tevő $x(t)$ folytonos térgörbe.

Ahhoz, hogy megkapjuk $x = x(1)$ -et, első lépésként a $[0,1]$ zárt intervallumot a következőképpen bontjuk fel:

$$0 = t_0 < t_1 < t_2 \dots < t_N = 1$$

és úgy tekintjük, hogy meg kell oldani a

$$H(x, t_i) = 0, \quad i = 1, \dots, N$$

problémát, valamilyen iterációs módszerrel, amely az $(i-1)$ -edik probléma x^{i-1} megoldását használja mint kezdő közelítést ahhoz, hogy az i -edik problémát megoldja. Ha $t_{i+1} - t_i$ megfelelően kicsi, akkor remélni lehet, hogy x^{i-1} megfelelően jó közelítés lesz x^i számára, tehát az iteráció konvergálni fog.

A következőkben az (1) egyenlet megoldásának egy a fentiekől különböző megközelítését ismertetjük.

Tegyük fel, hogy az $x: [0,1] \rightarrow D$ leképezés kielégíti a (2) feltételt folyamatosan differenciálható a $[0,1]$ zárt intervallumban és, hogy a H leképezés x és t szerint folytonosan differenciálható.

Akkor ha definiáljuk a következő függvényt

$$\Phi(t) = (H(x(t), t), t), \quad \forall t \in [0,1]$$

ez folytonosan differenciálható $[0,1]$ -en és a deriváltja:

$$\Phi'(t) = \frac{\partial H(x(t), t)}{\partial x} x'(t) + \frac{\partial H(x(t), t)}{\partial t}, \quad \forall t \in [0,1]$$

Miután $x = x(t)$ -ről feltettük, hogy kielégíti a (2) feltételt, ezért $\Phi'(t) = 0$ minden $t \in [0, 1]$ -re és ezért $x(t)$ kielégíti a következő differenciálegyenletet:

$$\frac{\partial H}{\partial x} x'(t) = - \frac{\partial H}{\partial t}, \quad \forall t \in [0, 1] \quad (3)$$

Azaz, ha $x: [0, 1] \rightarrow \mathbb{R}^n$ egy folytonos differenciálható megoldása a fenti differenciálegyenletnek és kielégíti a $H(x(0), 0) = 0$ kezdeti feltételt, akkor a közéértéktételből következik, hogy

$$\|H(x(t), t)\| = \|\Phi(t) - \Phi(0)\| \leq \sup_{0 \leq s \leq t} \|\Phi'(s)\| = 0$$

ugy, hogy $H(x(t), t) = 0$ bármilyen $t \in [0, 1]$ -re.

Ennél fogva a (3) -as differenciálegyenlet megoldása a $H(x(0), 0) = 0$ kezdeti feltétellel megadja az $Fx = 0$ egyenlet-rendszer megoldását.

A következő lépés tehát az, hogy meg kell oldani a (3) -as differenciálegyenletet valamilyen numerikus módszerrel. Ebből a célból átírjuk a differenciálegyenletet a következő alakba:

$$x'(t) = - \left(\frac{\partial H}{\partial x} \right)^{-1} \frac{\partial H}{\partial t}, \quad \forall t \in [0, 1] \\ H(x(0), 0) = 0 \quad (4)$$

Ezt az átalakítást mindig végre lehet hajtani, ha F Jacobi mátrixa nem szinguláris. Az egyik legegyszerűbb numerikus integrálási módszer az Euler módszer, ami a

$$x' = f(x, t), \quad x(0) = x^0, \quad t \in [0, 1]$$

differenciálegyenletre és a

$$0 < t_1 < t_2 < \dots \quad t_N = 1$$

particióra a következő alakú:

$$x^{k+1} = x^k + (t_{k+1} - t_k) f(x^k, t_k), \quad k = 0, 1, \dots, N-1$$

Ha alkalmazzuk a módszert a H leképezésre a következőt kapjuk:

$$x^{k+1} = x^k - (t_{k+1} - t_k) \frac{\partial H(x^k, t_k)^{-1}}{\partial x} \frac{\partial H(x^k, t_k)}{\partial t} \quad k = 0, 1, \dots, N-1$$

$H(x, t) = Fx + (t-1) Fx^0$ helyettesítés után a következő egyszerű alakot kapjuk:

$$x^{k+1} = x^k - h_k F'(x^k)^{-1} Fx^0, \quad k = 0, 1, \dots, N-1, \quad h_k = t_{k+1} - t_k$$

A tételben megadott feltételek mellett a differenciálegyenletnek van egy $x = x(t)$ folytonos megoldásgörbéje és ha h_k elég kicsi, akkor a számított x^k értékek megközelítik ezt a görbét és x^N megfelelően közeli kezdeti érték lesz ahhoz, hogy bármelyik iterációs módszer, például a Newton-módszer konvergáljon.

Ezt a $H(x, t) = 0$ megoldásának differenciálegyenletre való átalakítása módszerét először Davidenko vezette be, [5], ezért Davidenko-módszernek nevezik. A módszer nagyon elegáns és jól használható, de ennek is megvan az a hátránya, mint a Newton-módszernek, hogy alkalmazásához szükség van az F Jacobi mátrixának explicit kifejezésére. Ezért ha a Jacobi mátrix felírása nehézségbe ütközik, akkor inkább az elsőnek említett folytatásos módszerrel kell dolgozni.

Meg kell még említeni, hogy a differenciálegyenlet numerikus integrálásánál nem kötelező az Euler-módszert használni, lehet bármelyik magasabbrendű módszerrel is számolni, mint például a Runge-Kutta módszerrel, de mivel nem akarunk nagy pontosságra törekedni, az Euler-módszer is elegendőnek látszik.

3. NEMLINEÁRIS EGYENLETRENDSZEREK MEGOLDÁSÁRA KÉSZÜLT PROGRAMCSOMAG

A programcsomag valós együtthatós, legfeljebb 20 egyenletből álló 20 ismeretlenes nemlineáris egyenletrendszerek egy valós gyökének kiszámítására készült. Elkészítésénél arra törekedtünk, hogy ha az egyenletrendszernek van gyöke, akkor egy gyököt majdnem mindig ki is tudjunk számítani.

A nemlineáris egyenletrendszereket két csoportba osztottuk, az úgynevezett "polinom" alaku egyenletrendszerekre és a transzcendens egyenletrendszerekre. A polinom alaku egyenletrendszerek általános alakja a következő:

$$F x = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) \end{pmatrix}$$

ahol

$$f_i(x) = \sum_{j=1}^{k(i)} a_{ij} x_1^{m_1} x_2^{m_2} \dots x_n^{m_n}$$

Az ilyen alaku egyenletrendszerek Jacobi mátrixának explicit alakban való előállítása igen egyszerű, tehát ilyenkor ajánlatos a gyök megkeresését először a Newton-módszerrel megkísérelni, és csak akkor áttérni egy másik módszerre, ha a Newton-módszer nem konvergál. Kezdeti érték előállítására ilyenkor a Davidenko-féle differenciálegyenletes módszert lehet alkalmazni.

A programcsomag a polinom alaku egyenletrendszerek függvényértékének és Jacobi mátrixának előállítására is tartalmaz szubrutint, a felhasználónak csak az egyenletrendszert leíró állandókat kell a gépbe bevinni adatkártyákon.

Transzcendens egyenletrendszerek esetén természetesen a felhasználónak kell elkészíteni a függvényértékeket kiszámító szubrutint és ha a Newton-módszert szeretné alkalmazni, akkor a Jacobi mátrixot előállító szubrutint is. Transzcendens egyenleteknél a folytatásos módszert alkalmazzuk kezdőérték keresésére és előállítására. A módszer gyakorlatban való kipróbálásánál kiderült, hogy nem jó a $[0,1]$ intervallumot egyenlő lépésközökre felbontani, mert ha az $x(t)$ térgörbe folytonos is, általában nem egyenletesen folytonos, tehát bizonyos szakaszon elegendő sokkal nagyobb felosztás, míg lesznek olyan szakaszok, ahol sokkal kisebb felosztásra van szükség. Ezért a programban úgy módosítottuk a módszert, hogy elindulunk 0,1 hosszúságú lépésközökkel és addig haladunk így, amíg lehet. Amikor elérkezünk egy olyan ponthoz, hogy az előző szakaszban kapott kezdeti érték nem elég jó ahhoz, hogy a következő szakaszon a módszer konvergáljon, akkor a lépéshosszt megszorozzuk 10^{-1} -el és így próbálunk tovább számolni. Ha ez sem elég, akkor tovább szorozzuk 10^{-1} -el, stb. Ha ilyen kis lépéshosszal sikerül tovább lépni, akkor újra megnyújtjuk a lépéshosszt 0,1-re. Ilyen módon csak azokon a szakaszokon kell finom felosztást alkalmazni, ahol az valóban szükséges. A lépéshossz finomítást 10^{-4} -ig engedjük lemenni. Ha még ilyen finom felosztásnál sem tudunk tovább haladni, akkor abbahagyjuk a próbálkozást, mert feltehetőleg itt az $x(t)$ függvénynek szakadása van, tehát a módszert nem lehet alkalmazni.

A felhasználónak módjában van kezdőértékeket megadni adatkártyákon, de nem kötelező. Ha nem ad meg, akkor a program polinom alakú egyenletrendszer esetén, vagy olyankor amikor rendelkezésre áll a Jacobi mátrix, Davidenko-módszerrel, a többi esetben pedig a folytatásos módszerrel előállítja a kezdőértékeket. Ha a felhasználó által megadott kezdőértékekből a program nem tudott gyököt találni, akkor is alkalmazza a kezdőértéket számító eljárásokat és az így kapott értékekből újra megkísérli a gyök megkeresését. Ha a kezdőérték számító eljárások valamilyen okból nem működnek, akkor $[0,1]$ intervallumban egyenletes elosztású véletlen számokból kiindulva kísérel meg gyököt keresni.

A felhasználónak módjában áll hibakorlátot megadni, de ez nem kötelező. Ha nem kíván, akkor 0.0-t kell megadni hibakorlátként. Ilyenkor a program 10^{-6} -os hibakorláttal számol, azaz akkor tekint valamilyen x^k vektort a rendszer gyökének, ha két egymásutáni közelítő érték relatív távolsága, $|x^{k-1} - x^k|/|x^k|$ és a megfelelő $|F(x^k)|$ is kisebb, mint a hibakorlát. Ilyen módon kizárjuk, hogy a program egy stacionárius pontot, vagy egy nem nulla lokális minimumot gyökként elfogadjon.

Ha a felhasználó 10^{-6} -nál kisebb hibakorlátot ad meg és a program ilyen nagy pontosságra nem tudta a gyököket kiszámítani, akkor még 10^{-6} -al is megkísérli. Ennél kisebb hibakorlátot azonban nem ajánlatos előírni, mert a gép szóhossza miatt a többi jegy már gyakorlatilag kerekítési hibából adódik.

A programcsomagba a következő iterációs módszerek programja van beépítve:

- 1./ Newton-Raphson módszer.
- 2./ Kétlépéses Steffensen-módszer.
a 2.2-ben (1) alatt szereplő alappont megválasztással.
Iterációs képlete 2.2-ben (3).
- 3./ Kétlépéses Steffensen-módszer a 2.2-ben (2) alatt
szereplő alappont megválasztással.
- 4./ Seidel-módszer.
- 5./ Kétlépéses szelő módszer Wolfe-féle szelő formulával.
- 6./ Powell-féle hibrid módszer.
- 7./ n+1 lépéses szekvenciális szelő módszer Wolfe-féle
szelő formulával.

A felhasználónak módjában áll ezek közül a módszerek közül kiválasztani azt, amivel az egyenletrendszerét meg akarja oldani. Ha nem kíván választani, akkor polinom alaku egyenletrendszer esetén a Newton-módszert, transzcendens egyenletrendszer esetén pedig a kétlépéses szelő módszert alkalmazza a rendszer. Ha a kiválasztott módszer nem konvergál, akkor automatikusan áttér a Powell-módszerre és azzal is megkísérli a gyököket megkeresni. Természetesen ha azonnal a Powell-módszert alkalmaztuk, akkor ez

nem történik meg.

A programrendszer képes újraindítás nélkül tetszőleges számú polinom alakú és 4 transzcendens egyenletrendszert megoldani.

A programrendszer vezérlését egyenletrendszerenként egy paraméterkártya végzi. Ez a kártya a következő paramétereket tartalmazza:

N = Az egyenletrendszer rendszáma

I1 = $\begin{cases} 0 & \text{az egyenletrendszer polinom alakú} \\ 1, 2, 3, 4 & \text{az egyenletrendszer transzcendens} \end{cases}$

I2 = $\begin{cases} 0 & \text{nincs kezdeti érték} \\ 1 & \text{van kezdeti érték} \end{cases}$

I3 = $\begin{cases} 0 & \text{az utolsó feladatnál} \\ 1 & \text{még következik utána is feladat} \end{cases}$

I4 = $\begin{cases} 0 & \text{nem választ módszert} \\ 1 & \text{Newton-módszert választja} \\ 2 & \text{első Steffensen-módszert választja} \\ 3 & \text{második Steffensen-módszert választja} \\ 4 & \text{Seidel-módszert választja} \\ 5 & \text{kétlépéses szelő-módszert választja} \\ 6 & \text{Powell-módszert választja} \\ 7 & \text{n+1 lépéses szelő-módszert választja} \end{cases}$

A programrendszer egy állandóan a memóriában lévő főrészből és 9 overlay-ből áll. Az overlay-ken vannak a kezdőértéket számító programok és a 7 módszer programja. Ilyen módon mindig csak annak a módszernek a programja kerül be a memóriába, amire aktuálisan szükség van és ezért az egész programrendszer tulajdonképpen alig foglal el nagyobb belső memória területet, mint ha csak egy módszer programját tartalmazná. A rendszer igen könnyen bővíthető, mindössze egy utasítást kell kicsit megváltoztatni és két másik utasítást beszúrni a főprogramba ahhoz, hogy további módszer programját be tudjuk építeni.

Output-ként a sornyomtatón közli a mindig aktuális kezdőértékrendszert, az éppen alkalmazott módszer nevét, és ha a módszer konvergál, a kapott eredményeket, a gyök helyén vett helyettesítési értékeket és a gyök kiszámításához szükséges iterációs lépések számát a megfelelő módszernél. Ha a módszer nem konvergál ezt szöveggel jelzi. Amennyiben a gép által számított kezdeti értékből kiindulva 10^{-6} -os pontossággal Powell-módszerrel sem sikerült gyököt találni a következő szöveget közli:

AZ EGYENLETRENDSZER GYÖKÉT NEM LEHET KISZÁMITANI.

Ilyenkor egy lényegesen különböző kezdeti értékrendszerrel meg lehet újra kísérelni a gyök kiszámítását.

A programcsomag részletes kezelési utasítást a CDC 3300 Felhasználói ismertető 6. száma tartalmazza.

4. PRÓBA FELADATOK

A programcsomagot a következő feladatokon próbáltuk ki:

$$\begin{aligned} 1./ \quad & 3x^2y^3z + 2xy^2z^3 = 10 \\ & x^3y + z^2y^2 = 3 \\ & 6y^2z^2 + 4xyz + 3xyz^2 = 4 \end{aligned}$$

$$\begin{aligned} 2./ \quad & \sin x - y = 1.32 \\ & \cos y - x = -0.85 \end{aligned}$$

$$\begin{aligned} 3./ \quad & x_1^2 - 2\cos(x_2) + 9 + x_3^2 - 11 = 0 \\ & 3x_1 + 2x_2 - 1.5x_3 - 2 = 0 \\ & 8x_3 + 2^{x_1 - x_2} + x_1x_3 - (x_1 + x_2 + x_3)^2 + 7 = 0 \end{aligned}$$

$$\begin{aligned} 4./ \quad & x_1 + x_2 - \sin x_3 + x_4 - 1.5 = 0 \\ & 2x_1^2 - x_2 + \cos x_3 - 3/2 = 0 \\ & x_1x_2 + x_4 - 1 = 0 \\ & 5x_1 - x_2 - \sin 3x_3 - 2 = 0 \end{aligned}$$

$$\begin{aligned} 5./ \quad & 2x_1 + \log x_2 + \arctg x_3 - x_4^2 + x_5 - 3.90675 = 0 \\ & 2x_1x_2 - 4x_2 + x_5^2 + 35.711 = 0 \\ & x_3/x_5 - x_3 + x_4 \log x_2 - 0.785398 = 0 \\ & 3x_1 + x_2 - x_3 + x_1^5 - 13.42265 = 0 \\ & 3x_2 - \tg x_4 + \log x_1 - 29 = 0 \end{aligned}$$

Az 1./ egyenletrendszer futtatásánál a következő eredményeket kaptuk:

Davidenko módszere a következő kezdeti értékrendszerrel eredményezte:

$$\begin{aligned} \text{te:} \quad & x_1 = 1.0350877 \\ & x_2 = 2.4721247 \\ & x_3 = 0.20394566 \end{aligned}$$

Ebből a kezdeti rendszerből a Newton-módszer 10^{-6} -os hibakorláttal a következő gyökrendszert találta:

$$x_1=1.03792421$$

$$x_2=2.45106599$$

$$x_3=0.20777668$$

A helyettesítési értékek ebben a pontban (kerekítve):

$$f_1=-0.7 \cdot 10^{-9}$$

$$f_2=0.11 \cdot 10^{-9}$$

$$f_3=-0.46 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 3

Ugyanabból a kezdeti értékrendszerből az első alappont választásos Steffensen-módszerrel:

$$x_1=1.03792421$$

$$x_2=2.45106599$$

$$x_3=0.207776686$$

Helyettesítési értékek:

$$f_1=-0.186 \cdot 10^{-8}$$

$$f_2=-0.17 \cdot 10^{-9}$$

$$f_3=-0.46 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 4

A második alappont választásos Steffensen-módszerrel:

$$x_1=1.03792421$$

$$x_2=2.45106599$$

$$x_3=0.20777668$$

Helyettesítési értékek:

$$f_1 = -0.69 \cdot 10^{-9}$$

$$f_2 = -0.12 \cdot 10^{-9}$$

$$f_3 = 0.81 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 4

Powell-féle hibrid módszerrel:

$$x_1 = 1.03792418$$

$$x_2 = 2.45106602$$

$$x_3 = 0.207776679$$

Helyettesítési értékek:

$$f_1 = -0.64 \cdot 10^{-6}$$

$$f_2 = -0.25 \cdot 10^{-6}$$

$$f_3 = -0.22 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 9

Kétlépéses szelő módszerrel:

$$x_1 = 1.03792412$$

$$x_2 = 2.45106629$$

$$x_3 = 0.207776647$$

Helyettesítési értékek:

$$f_1 = 0.97 \cdot 10^{-7}$$

$$f_2 = -0.38 \cdot 10^{-6}$$

$$f_3 = -0.63 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 20

(n+1) -lépéses szelő módszerrel:

$$x_1 = 1.03792422$$

$$x_2 = 2.45106600$$

$$x_3 = 0.20777669$$

Helyettesítési értékek:

$$f_1 = 0.49 \cdot 10^{-6}$$

$$f_2 = 0.85 \cdot 10^{-7}$$

$$f_3 = 0.15 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 14

A 2./ egyenletrendszer futtatásánál a következő eredményeket kaptuk:

A Davidenko-módszerrel kapott kezdeti értékrendszer:

$$x_1 = 1.792234$$

$$x_2 = -0.3440346$$

Ebből a kezdeti értékrendszerből Newton-módszerrel, 10^{-6} -os pontossággal kapott eredmények:

$$x_1 = 1.7913386$$

$$x_2 = -0.344221036$$

Helyettesítési értékek:

$$f_1 = 0.29 \cdot 10^{-10}$$

$$f_2 = 0.0$$

A szükséges iterációs lépések száma: 2

A folytatásos módszerrel kapott kezdeti értékrendszer:

$$x_1 = 1.7948$$

$$x_2 = -0.34477$$

Steffensen 1 módszerrel kapott eredmények:

$$x_1 = 1.791338$$

$$x_2 = -0.344221$$

Helyettesítési értékek:

$$f_1 = 0.0$$

$$f_2 = -0.116 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 2

Steffensen 2 módszerrel kapott eredmények:

$$x_1 = 1.7913386$$

$$x_2 = -0.344221036$$

Helyettesítési értékek:

$$f_1 = 0.29 \cdot 10^{-10}$$

$$f_2 = 0.0$$

A szükséges iterációs lépések száma: 2

Powell-módszerrel kapott eredmények:

$$x_1 = 1.7913434$$

$$x_2 = -0.344199$$

Helyettesítési értékek:

$$f_1 = -0.23 \cdot 10^{-4}$$

$$f_2 = 0.25 \cdot 10^{-5}$$

Iterációs lépések száma: 2

Kétlépéses szelő módszerrel kapott eredmények:

$$x_1 = 1.7913384$$

$$x_2 = -0.34422076$$

Helyettesítési értékek:

$$f_1 = -0.23 \cdot 10^{-6}$$

$$f_2 = 0.27 \cdot 10^{-6}$$

Szükséges iterációs lépések száma: 5

(n+1) -lépéses szelő módszerrel kapott eredmények:

$$x_1 = 1.7913386$$

$$x_2 = -0.344221036$$

Helyettesítési értékek:

$$f_1 = -0.29 \cdot 10^{-10}$$

$$f_2 = 0.17 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 3

A 3./ egyenletrendszer futtatásánál a következő eredményeket kaptuk:

A folytatásos módszerrel kapott kezdeti értékrendszer:

$$x_1 = 1.9997929$$

$$x_2 = 1.0001595$$

$$x_3 = 3.9997984$$

Kétlépéses szelő módszerrel kapott eredmények:

$$x_1 = 1.99999990$$

$$x_2 = 1.00000008$$

$$x_3 = 3.99999990$$

Helyettesítési értékek:

$$f_1 = -0.5 \cdot 10^{-6}$$

$$f_2 = 0.0$$

$$f_3 = 0.68 \cdot 10^{-7}$$

A szükséges iterációs lépések száma: 9

Steffensen 1 módszerrel kapott eredmények:

$$x_1 = 2.00000000$$

$$x_2 = 1.00000001$$

$$x_3 = 4.00000001$$

Helyettesítési értékek:

$$f_1 = 0.0$$

$$f_2 = -0.46 \cdot 10^{-9}$$

$$f_3 = -0.26 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 4

Steffensen 2 módszerrel kapott eredmények:

$$x_1 = 2.00000000$$

$$x_2 = 0.999999986$$

$$x_3 = 3.99999999$$

Helyettesítési értékek:

$$f_1 = 0.23 \cdot 10^{-9}$$

$$f_2 = -0.17 \cdot 10^{-9}$$

$$f_3 = -0.24 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 3

Powell-módszerrel kapott eredmények:

$$x_1 = 1.99999993$$

$$x_2 = 1.00000005$$

$$x_3 = 3.99999993$$

Helyettesítési értékek:

$$f_1 = -0.32 \cdot 10^{-6}$$

$$f_2 = 0.0$$

$$f_3 = 0.41 \cdot 10^{-7}$$

A szükséges iterációs lépések száma: 7

A 4./ egyenletrendszer futtatásánál a következő eredményeket kaptuk:

A folytatásos módszerrel kapott kezdeti értékrendszer:

$$x_1 = 0.76457135$$

$$x_2 = 0.83166963$$

$$x_3 = 0.47838086$$

$$x_4 = 0.36412966$$

Steffensen 1 módszerrel kapott eredmények:

$$x_1=0.764613537$$

$$x_2=0.832183457$$

$$x_3=0.478556432$$

$$x_4=0.363701264$$

Helyettesítési értékek:

$$f_1=0.2 \cdot 10^{-9}$$

$$f_2=-0.35 \cdot 10^{-9}$$

$$f_3=-0.29 \cdot 10^{-10}$$

$$f_4=-0.93 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 2

Steffensen 2 módszerrel kapott eredmények:

$$x_1=0.764613537$$

$$x_2=0.832183458$$

$$x_3=0.478556433$$

$$x_4=0.363701263$$

Helyettesítési értékek:

$$f_1=0.0$$

$$f_2=-0.2 \cdot 10^{-9}$$

$$f_3=0.32 \cdot 10^{-9}$$

$$f_4=0.17 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 2

Powell-módszerrel kapott eredmények:

$$x_1=0.7646135$$

$$x_2=0.83218342$$

$$x_3=0.47855642$$

$$x_4=0.36370132$$

Helyettesítési értékek:

$$f_1 = -0.66 \cdot 10^{-8}$$

$$f_2 = -0.63 \cdot 10^{-7}$$

$$f_3 = 0.44 \cdot 10^{-9}$$

$$f_4 = -0.13 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 8

Kétlépéses szelő módszerrel kapott eredmények:

$$x_1 = 0.764613421$$

$$x_2 = 0.832182734$$

$$x_3 = 0.478556198$$

$$x_4 = 0.363701913$$

Helyettesítési értékek:

$$f_1 = 0.18 \cdot 10^{-7}$$

$$f_2 = 0.48 \cdot 10^{-6}$$

$$f_3 = -0.29 \cdot 10^{-10}$$

$$f_4 = 0.24 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 8

(n+1)-lépéses szelő módszerrel kapott eredmények:

$$x_1 = 0.764613447$$

$$x_2 = 0.832183360$$

$$x_3 = 0.478556376$$

$$x_4 = 0.36370122$$

Helyettesítési értékek:

$$f_1 = -0.18 \cdot 10^{-6}$$

$$f_2 = -0.15 \cdot 10^{-6}$$

$$f_3 = -0.19 \cdot 10^{-6}$$

$$f_4 = -0.33 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 7

Az 5./ -ös egyenletrendszer futtatásánál a következő eredményeket kaptuk:

Adatkártyákról bevitt kezdetiérték rendszer:

$$x_1=1.5$$

$$x_2=7.2$$

$$x_3=0.18$$

$$x_4=0.6$$

$$x_5=1.5$$

Ezekből a kezdetiértékekből a kezdetiértékekből Newton-módszerrel a következő eredményeket kaptuk:

$$x_1=1.00000343$$

$$x_2=9.99998493$$

$$x_3=0.577348668$$

$$x_4=0.785376753$$

$$x_5=0.999962841$$

Helyettesítési értékek:

$$f_1=-0.15 \cdot 10^{-7}$$

$$f_2=-0.18 \cdot 10^{-8}$$

$$f_3=-0.3 \cdot 10^{-6}$$

$$f_4=0.23 \cdot 10^{-9}$$

$$f_5=-0.89 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 39

Ugyanabból a kezdeti értékrendszerből Powell-módszerrel kapott eredmények:

$$x_1=1.00000339$$

$$x_2=9.99998565$$

$$x_3=0.577349214$$

$$x_4=0.785377384$$

$$x_5=0.999963453$$

Helyettesítési értékek:

$$f_1=0.36 \cdot 10^{-7}$$

$$f_2=0.93 \cdot 10^{-9}$$

$$f_3=-0.47 \cdot 10^{-8}$$

$$f_4=0.69 \cdot 10^{-9}$$

$$f_5=-0.33 \cdot 10^{-7}$$

A szükséges iterációs lépések száma: 26

Adatkártyákról bevitt más kezdeti értékrendszer:

$$x_1=0.75$$

$$x_2=9.77$$

$$x_3=-0.6$$

$$x_4=0.56$$

$$x_5=0.98$$

Ebből a kezdeti értékrendszerből Steffensen 2 módszerrel kapott eredmények:

$$x_1=1.00000340$$

$$x_2=9.99998565$$

$$x_3=0.577349254$$

$$x_4=0.785377397$$

$$x_5=0.999963468$$

Helyettesítési értékek:

$$f_1=-0.29 \cdot 10^{-9}$$

$$f_2=0.93 \cdot 10^{-9}$$

$$f_3=0.58 \cdot 10^{-10}$$

$$f_4 = -0.23 \cdot 10^{-9}$$

$$f_5 = 0.46 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 6

Kétlépéses szelő módszerrel kapott eredmények:

$$x_1 = 1.00000341$$

$$x_2 = 9.99998507$$

$$x_3 = 0.577349316$$

$$x_4 = 0.785377524$$

$$x_5 = 0.999963524$$

Helyettesítési értékek:

$$f_1 = -0.66 \cdot 10^{-7}$$

$$f_2 = 0.4 \cdot 10^{-6}$$

$$f_3 = 0.95 \cdot 10^{-7}$$

$$f_4 = -0.23 \cdot 10^{-9}$$

$$f_5 = -0.23 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 33

Powell-módszerrel kapott eredmények:

$$x_1 = 1.00000340$$

$$x_2 = 9.99998567$$

$$x_3 = 0.577349272$$

$$x_4 = 0.785377401$$

$$x_5 = 0.999963472$$

Helyettesítési értékek:

$$f_1 = 0.12 \cdot 10^{-7}$$

$$f_2 = -0.14 \cdot 10^{-7}$$

$$f_3 = 0.32 \cdot 10^{-8}$$

$$f_4 = -0.16 \cdot 10^{-8}$$

$$f_5 = 0.18 \cdot 10^{-7}$$

A szükséges iterációs lépések száma: 23

Folytatásos módszerrel nyert kezdeti értékrendszer:

$$x_1 = 0.99999266$$

$$x_2 = 9.9999076$$

$$x_3 = 0.57722837$$

$$x_4 = 0.78516793$$

$$x_5 = 0.99981400$$

Steffensen 2 módszerrel kapott eredmények:

$$x_1 = 1.00000340$$

$$x_2 = 9.99998566$$

$$x_3 = 0.577349254$$

$$x_4 = 0.785377397$$

$$x_5 = 0.999963468$$

Helyettesítési értékek:

$$f_1 = 0.0$$

$$f_2 = 0.93 \cdot 10^{-9}$$

$$f_3 = 0.26 \cdot 10^{-9}$$

$$f_4 = 0.46 \cdot 10^{-9}$$

$$f_5 = -0.46 \cdot 10^{-9}$$

A szükséges iterációs lépések száma: 2

Kétlépéses szelő módszerrel kapott eredmények:

$$x_1 = 1.00000336$$

$$x_2 = 9.99998621$$

$$x_3 = 0.577349645$$

$$x_4=0.785377944$$

$$x_5=0.999964158$$

Helyettesítési értékek:

$$f_1=0.67 \cdot 10^{-7}$$

$$f_2=0.71 \cdot 10^{-6}$$

$$f_3=0.17 \cdot 10^{-6}$$

$$f_4=0.0$$

$$f_5=0.53 \cdot 10^{-6}$$

A szükséges iterációs lépések száma: 15

Powell-módszerrel kapott eredmények:

$$x_1=1.00000340$$

$$x_2=9.99998567$$

$$x_3=0.577349264$$

$$x_4=0.785377410$$

$$x_5=0.999963479$$

Helyettesítési értékek:

$$f_1=-0.19 \cdot 10^{-8}$$

$$f_2=0.15 \cdot 10^{-7}$$

$$f_3=0.73 \cdot 10^{-8}$$

$$f_4=0.0$$

$$f_5=0.37 \cdot 10^{-8}$$

A szükséges iterációs lépések száma: 9

I R O D A L O M

- [1] M.Altman: A generalization of Newton's method.
Bull.Acad.Polon.Sci.Ser.Sci.Math. Astronom.Phys. 3,
189-193. [1955]
- [2] C.G.Broyden: Recent developments in solving nonlinear
algebraic system. Numerical methods for nonlinear alge-
braic equations. 61-73. [1970]
- [3] D.Davidenko: On a new method of numerically integrating
a system of nonlinear equations. Dokl.Acad.Nauk. SSSR 88,
601-604. [1953]
- [4] D.Davidenko: On the approximate solution of a system of
nonlinear equations. Ukrain.Mat. Z. 5, 196-206. [1953]
- [5] D.F.Davidenko: An application of the method of variation
of parameters to the construction of iterative formula
of increased accuracy for numerical solutions of nonlin-
ear equations. Soviet Math.Dokl. 6. 702-706. [1965]
- [6] K.Levenberg: A method for the solution of certain non-
linear problems in least squares. Quart.Appl.Math.2.
164-168. [1944]
- [7] D.Marquardt: An algorithm for least squares estimation
of nonlinear parameters. SIAM J. Appl.Math.11. 431-441.
[1963]
- [8] G.H.Meyer: On solving nonlinear equations with a one-
parameter operator imbedding. SIAM J. Numer.Anal.5.
739-752. [1967]
- [9] J.Ortega and W.Rheinboldt: Iterative solution of non-
linear equation in several variables. Academic Press.
[1970]
- [10] M.J.D.Powell: A hybrid method for nonlinear equations.
Numerical methods for nonlinear algebraic equations.
87-114. [1970]

- [11] I.J.Steffensen: Remarks on iteration. Skand.
Aktuarietidskr.16, 67-72. [1933]

- [12] P.Wolfe: The secant method for simultaneous nonlinear
equations. Counn.ACM.2, 12-13. [1959]

A TANULMÁNYOK sorozatban eddig megjelentek:

- 1/1973 Pásztor Katalin: Módszerek Boole-függvények minimális vagy nem redundáns, $\{\wedge, \vee, \neg\}$ vagy $\{\text{NOR}\}$ vagy $\{\text{NAND}\}$ bázisbeli, zárójeles vagy zárójel nélküli formuláinak előállítására
- 2/1973 Вашкеви Иштван: Расчленение многосвязных промышленных процессов с помощью вычислительных машины
- 3/1973 Ádám György: A számítógépipar helyzete 1972 második felében
- 4/1973 Bányász Csilla: Identification in the Presence of Drift
- 5/1973* Gyürki J. - Laufer J. - Grint M. - Somló J.: Optimalizáló adaptív szerszámgépirányítási rendszerek
- 6/1973 Szelke Erzsébet - Tóth Károly: Felhasználói Kézikönyv /USER MANUAL/ a Folytonos Rendszerek Szimulációjára készült ANDISIM programnyelvhez
- 7/1973 Legendi Tamás: A CHANGE nyelv/multiprocesszor
- 8/1973 Klafszky Emil: Geometriai programozás és néhány alkalmazása
- 9/1973 R. Narasimhan: Picture Processing Using Pax
- 10/1973 Dibuz Ágoston - Gáspár János - Várszegi Sándor: MANU-WRAP hátlaphuzalozó. MSI-TESTER integrált áramköröket mérő, TESTOMAT-C logikai hálózatokat vizsgáló berendezések ismertetése
- 11/1973 Matolcsi Tamás: Az optimum-számítás egy új módszeréről
- 12/1973 Makroprocesszorok, programozási nyelvek. Cikkgyűjtemény az NJSzT és SzTAKI közös kiadásában.
Szerkesztette: Legendi Tamás
- 13/1973 Jedlovsky Pál: Új módszer bonyolult retifikáló oszlopok vegyészmérnöki számítására
- 14/1973 Bakó András: MTA Kutatóintézeteinek bérszámfejtése számítógéppel
- 15/1973 Ádám György: Kelet-nyugati kapcsolatok a számítógépiparban

- 16/1973 Fridrich Ilona - Uzsoky Miklós: LIDI-72 LIstakezelő rendszer a Digitális Osztályon, 1972. évi változat
- 17/1974 Gyürki József: Adaptív termelésprogramozó rendszer /APS/ termelő műhelyek irányítására
- 18/1974 Pikler Gyula: MINI-Számítógépes interaktív alkatrészprogramíró rendszer NC szerszámgépek automatikus programozásához
- 19/1974 Gertler, J. - Sedlak, J.: Software for process control
- 20/1974 Vámos, T. - Vassy, Z.: Industrial Pattern Recognition Experiment-A Syntax Aided Approach
- 21/1974 A KGST I.-15-1.: Diszkrét rendszerek automatikus tervezése c. témában 1973. februárban rendezett szeminárium előadásai
- 22/1974 Arató M. - Benczúr, A. - Krámlí, A. - Pergel, J.: Stochastic Processes, Part I.
- 23/1974 Benkó Sándor - Renner Gábor: Erősen telített mágneses körök számítógépes tervezési módszere
- 24/1974 Kovács György - Franta Lászlóné: Programcsomag elektronikus berendezések hátlapuzalozásának tervezésére
- 25/1974 Járdán R. Kálmán: Háromfázisu tirisztoros inverték állandósult tranziens jelenségei és belső impedanciája
- 26/1974 Gergely József: Numerikus módszerek sparse mátrixokra
- 27/1974 Somló János: Analitikus optimalizálás
- 28/1974 Vámos Tibor: Tárgyfelismerési kísérlet nyelvi módszerekkel
- 29/1974 Mórítz Péter: Vegyészmérnöki számítási módszerek fázis-egyensúlyok és kémiai egyensúlyok vizsgálatára
- 30/1974 Vámos, T. - Vassy, Z.: THE BUDAPEST ROBOT-Pragmatic intelligence -.
- 31/1975 Nagy István: Frekvenciásos, középfrekvenciás inverter elmélete

- 32/1975 Singer - Borossay - Koltai: Gázhálózatok optimális irányítása különös tekintettel a Fővárosi Gázművek hálózataira
- 33/1975 Vámos Tibor - Vassy Zoltán: Limited and Pragmatic Robot Intelligence
- L.Mérő - Z.Vassy: A Simplified and Fastened Version of the Hueckel Operator for Finding Optimal Edges in Pictures
- Галло В.: Программа для распознавания геометрических образов, основанная на лингвистическом методе описания и анализа геометрических структур
- 34/1975 Nemes László: Pattern Identification Method for Industrial Robots by Extracting the main Features of Objects
- 35/1975 Garádi János - Krámlí András - Ratkó István - Ruda Mihály: Statisztikai és számítástechnikai módszerek alkalmazása kórházi morbiditás vizsgálatokban
- 36/1975 Renner Gábor: Elektromágneses tér számítása nagyhőmérsékletű anyagban
- 37/1975 Edgardo Felipe: Specification problems of a process control display

757588 MTA KESZ Sokszorosító. F. v.: Szabó Gyula



